

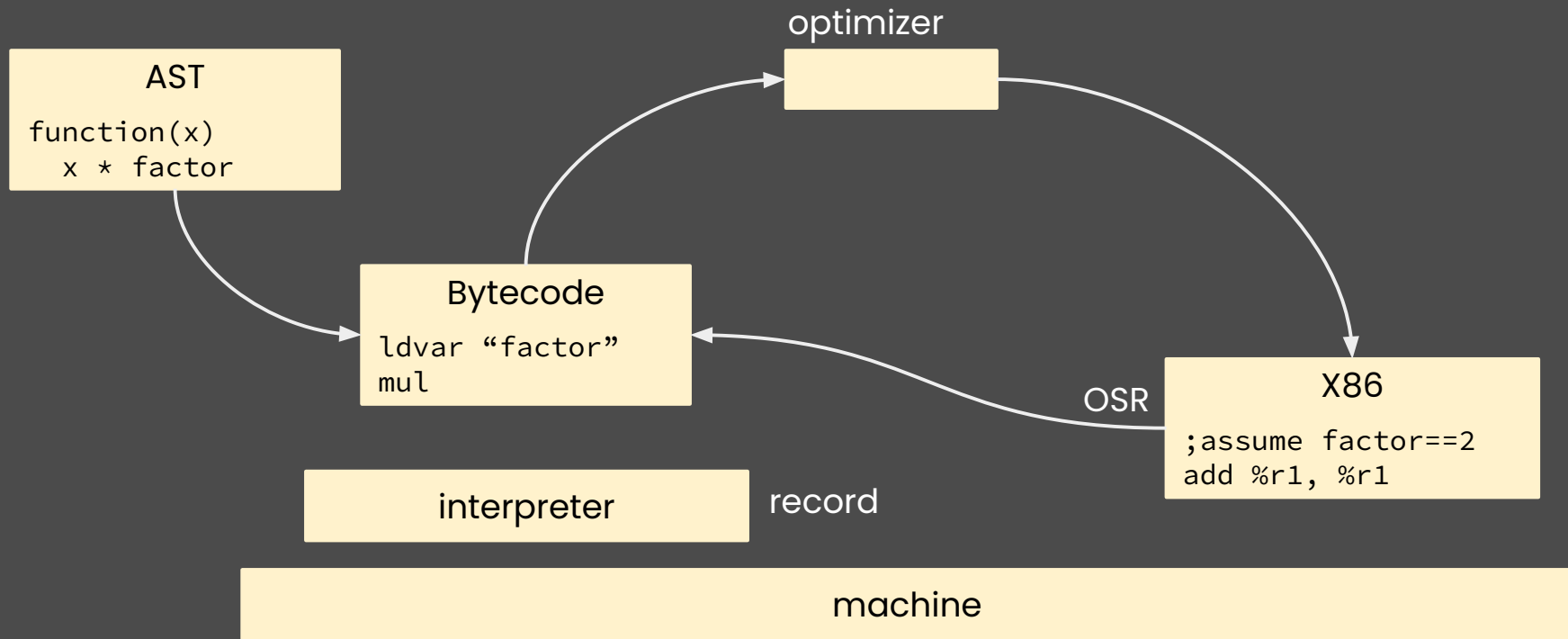
Deoptless

Speculation with Dispatched On-Stack Replacement
and Specialized Continuations

Olivier Flückiger, Jan Ječmen, Sebastián Krynski, Jan Vitek

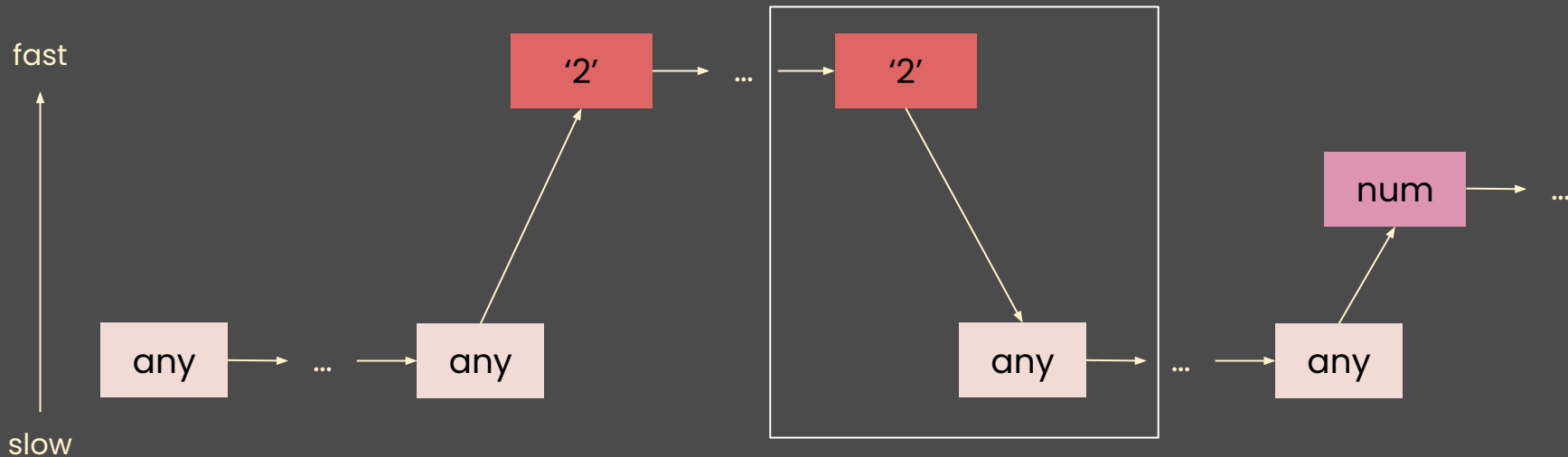
PLDI — 6/15/22

A typical JIT

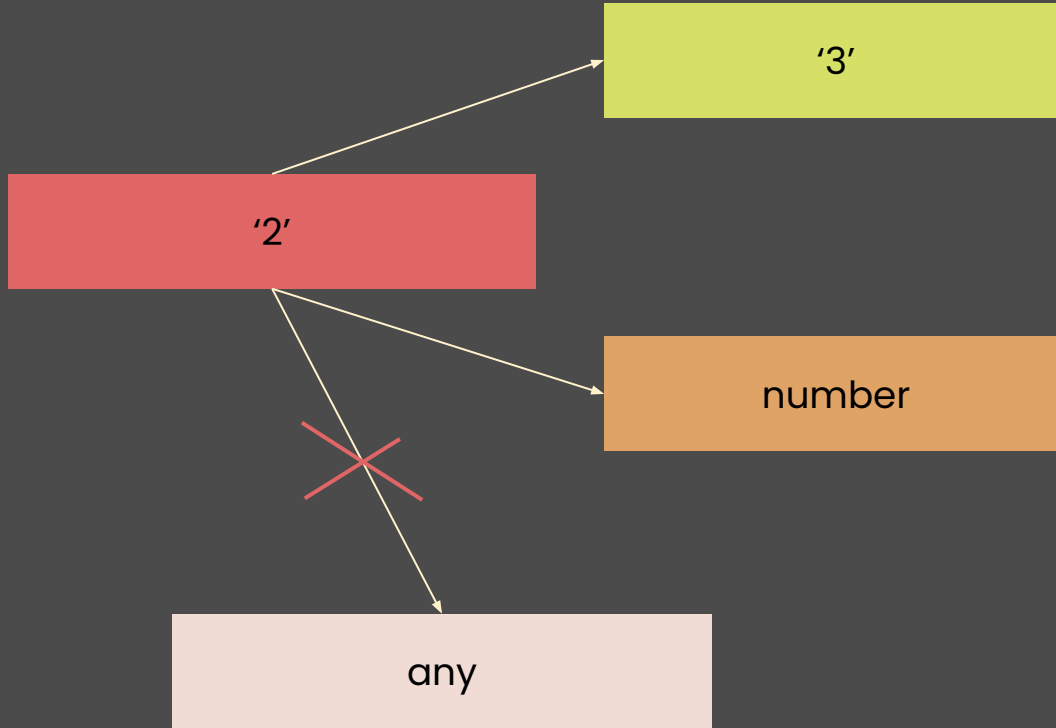


Speculation: Opt, fail, rinse and repeat

```
scale ← function(x) {  
  # assume factor == ?  
  x * factor  
}
```

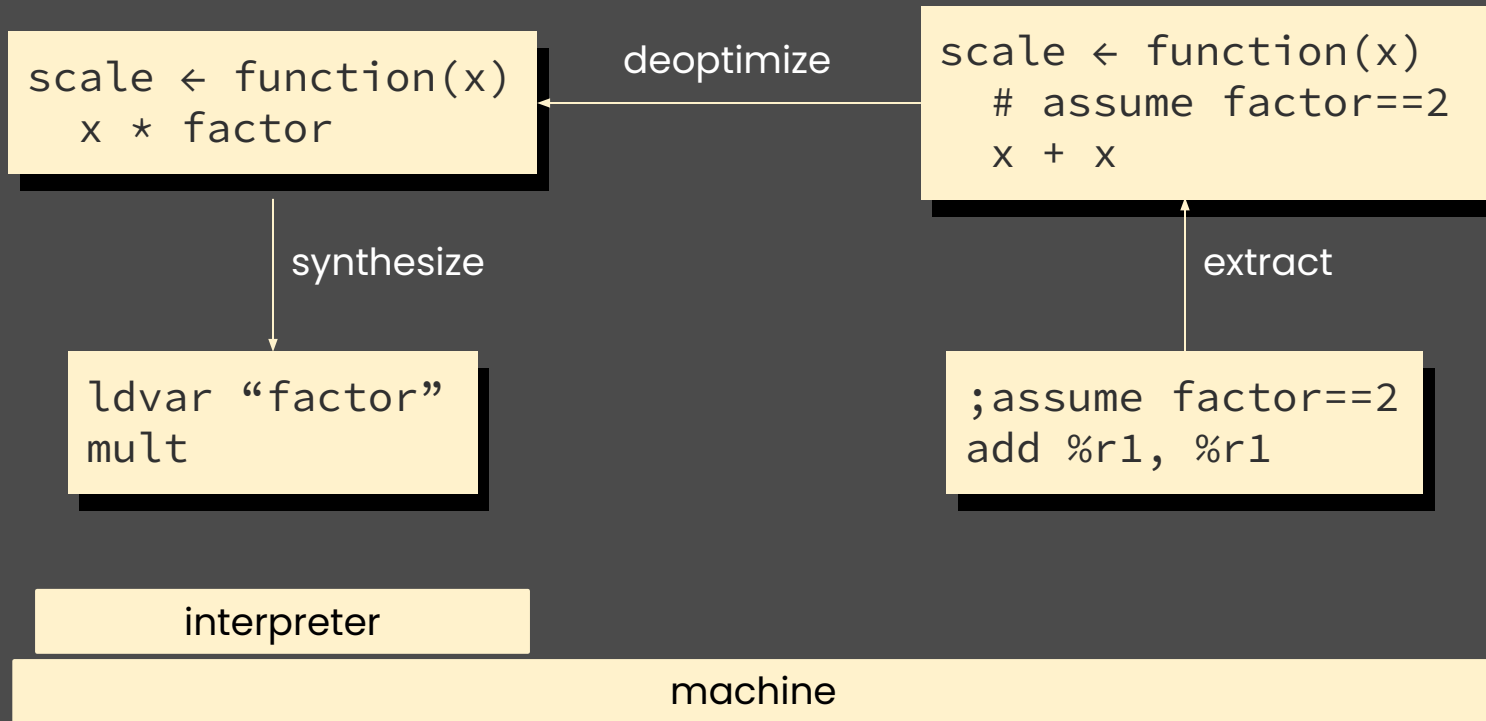


Deoptless

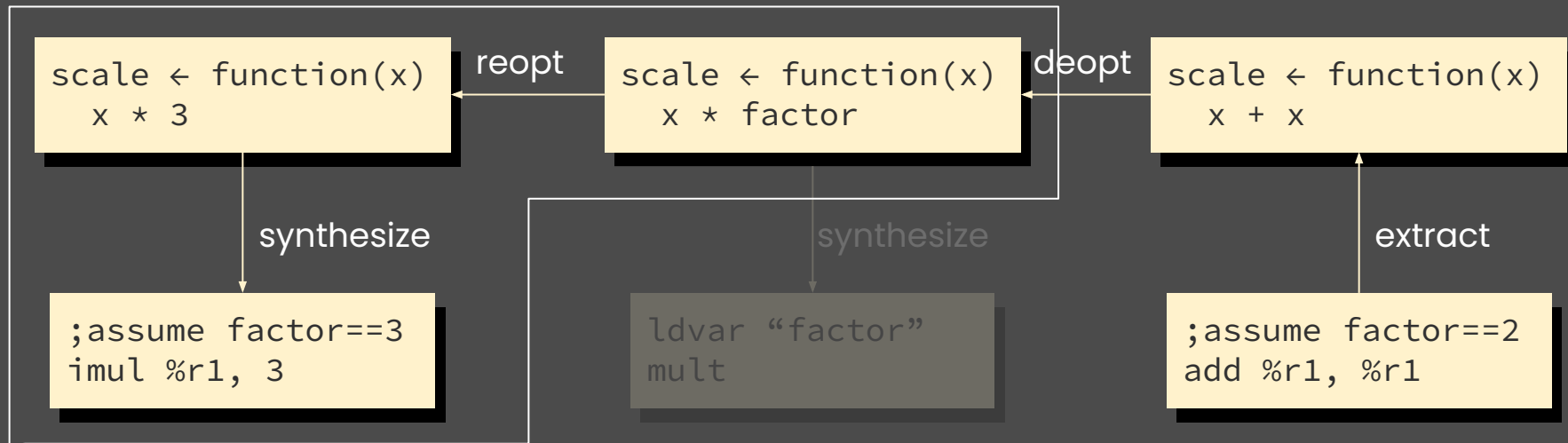


```
scale ← function(x) {  
  # assume factor == ?  
  x * factor  
}
```

On-Stack Replacement (OSR)

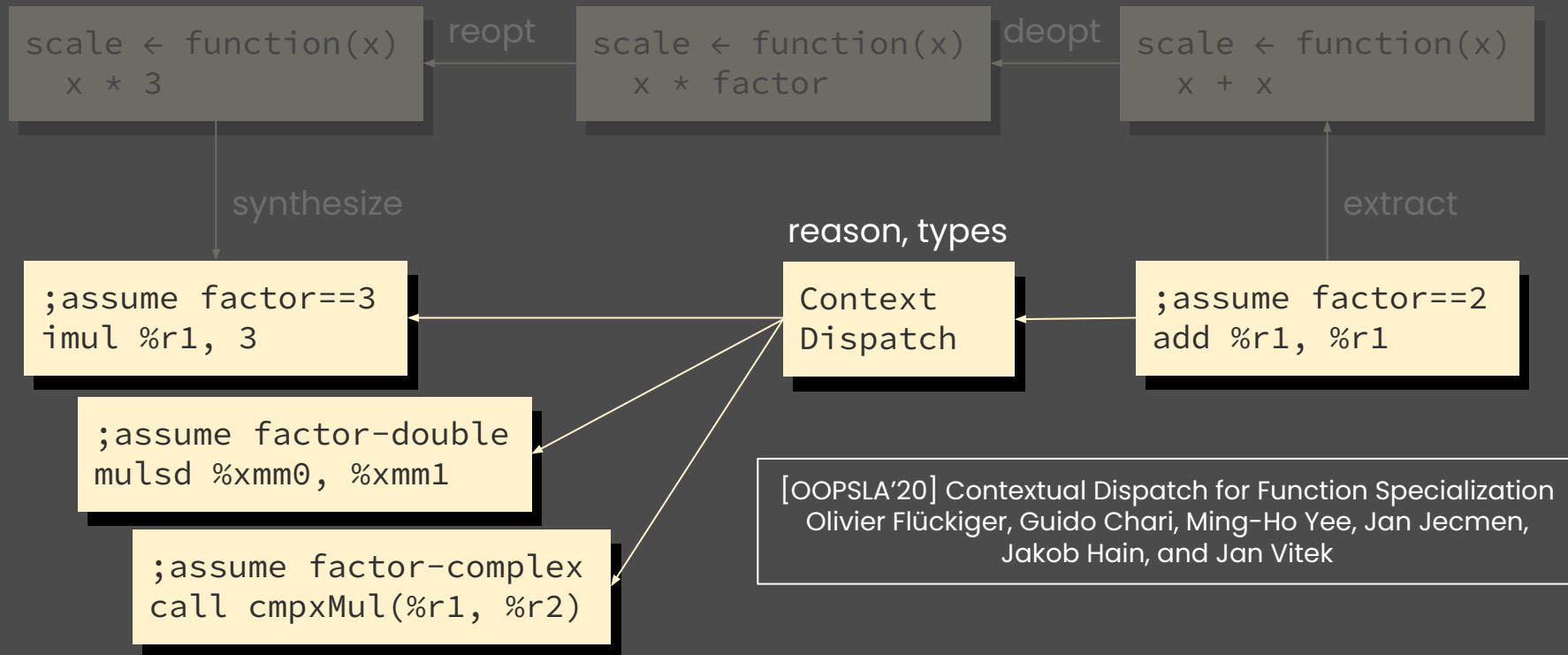


Optimized-to-Optimized OSR



OSR-in

Deoptless

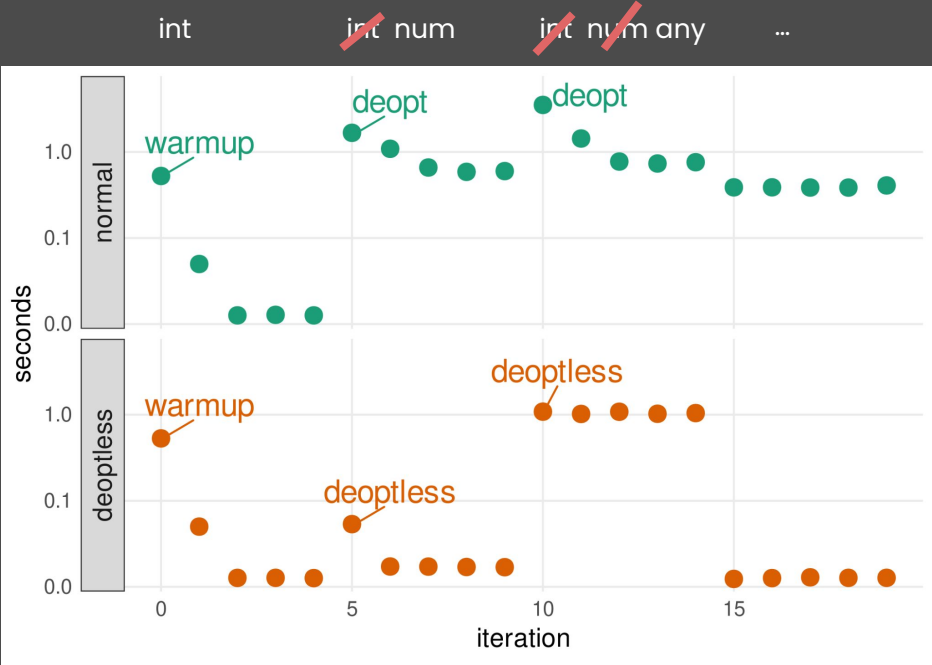


Example

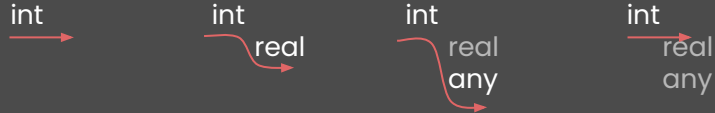
```

sum ← function() {
  total ← 0
  for (i in 1:length)
    total ← total + data[[i]]
  total
}

```

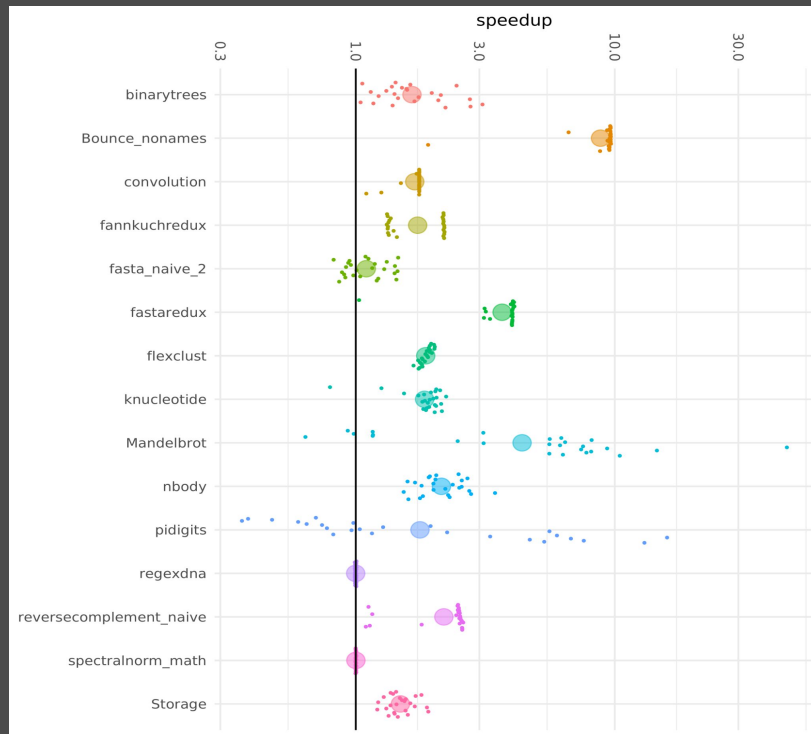


int ~~int num~~ ~~int num any~~ ...



Experiments

- Example programs
- Compare with sampling-based approach
- Synthetic benchmarks



Handling Random Deopts

OSR + CD = Deoptless

```
scale ← function(x)  
x * factor
```

OSR-in

n:1

OSR-out

```
;assume factor==3
```

```
Context  
Dispatch
```

```
;assume factor==2
```

```
;assume factor-double
```

```
;assume factor-complex
```